Proceedings of DSCC 2014
ASME 2014 Dynamic Systems and Control Conference
October 22-24, 2014, San Antonio, TX, USA

DSCC2014-6365

SCALABLE ANOMALY DETECTION AND ISOLATION IN CYBER-PHYSICAL SYSTEMS USING BAYESIAN NETWORKS

Sudha Krishnamurthy*
Email:krishnadots@gmail.com

Soumik Sarkar, Ashutosh Tewari
United Technologies Research Center
Email: {sarkars@utrc.utc.com, tewaria@utrc.utc.com}

ABSTRACT

Anomalies in cyber-physical systems may arise due to malicious cyber attacks or operational faults in the physical devices. Accurately detecting the anomalies and isolating their root-causes is important for identifying appropriate reactive and preventive measures and building resilient cyber-physical systems. Anomaly detection and isolation in cyber-physical systems is challenging, because the impact of a cyber attack on the operation of a physical system may manifest itself only after some time. In this paper, we present a Bayesian network approach for learning the causal relations between cyber and physical variables as well as their temporal correlations from unlabeled data. We describe the data transformations that we performed to deal with the heterogeneous characteristics of the cyber and physical data, so that the integrated dataset can be used to learn the Bayesian network structure and parameters. We then present scalable algorithms to detect different anomalies and isolate their respective root-cause using a Bayesian network. We also present results from evaluating our algorithms on an unlabeled dataset consisting of anomalies due to cyber attacks and physical faults in a commercial building system.

INTRODUCTION

Cyber-physical systems (CPS) combine computing and communication capabilities with monitoring and control of entities in the physical world. CPS systems are part of many safety-critical infrastructures and industrial control systems, such as electric power grids and building automation systems. Traditional approaches for protecting control systems have primarily focused on gradual deterioration or abrupt faults in physical components. However, the coupling between information and communication technologies and the physical controllers in a CPS system makes the control system more vulnerable, especially since networked systems make it possible to launch remote attacks. Hence, there is a growing need for protecting control systems against malicious cyber attacks. As part of cyber-security mechanisms, several authentication and access control technologies have been developed for protecting information. These technologies can also be used to prevent attacks in cyber-physical control systems to some extent. However, in addition, a resilient CPS architecture needs to include mechanisms for detecting and reacting to anomalies.

Anomaly detection refers to the problem of finding patterns that do not conform to expected behavior. Traditional anomaly detection schemes for cyber security analyze network traces for detecting network anomalies, but do not analyze the impact of attacks on physical components. On the other hand, system theory focuses more on reliability and stability of physical systems, but does not completely model information technology (IT) infrastructure. Prior work in fault tolerant control systems use redundancy and reconfiguration mechanisms to address the vulnerability of sensors and actuators to physical failures [1]. These techniques primarily focus on reliability and do not address vulnerabilities arising from security attacks. Recently, in [2], the authors suggest that the physical controllers can be monitored to detect anomalies that cannot be detected through IT mechanisms. Likewise, in [3], the authors provide some ways of leveraging system-theoretic techniques to counter cyber security attacks, in the context of a smart power grid. In [4], a smart power grid

*The authors would like to thank United Technologies Research Center for supporting this work.

is modeled as an undirected graph and a polynomial-time detection algorithm based on generalized likelihood ratio with L_1 norm regularization is used for finding small, but unobservable attacks. However, in order to successfully detect CPS anomalies and perform root-cause analysis to establish whether the anomalies are a result of a cyber attack or a fault in the physical components (sensors, actuators, controllers), we need an integrated approach that is based on understanding the cause-effect relationship between the cyber components and the physical system.

In this paper, we propose an anomaly detection method that relies on a probabilistic graphical model of the underlying CPS. Specifically, we use a Bayesian network to characterize a CPS under nominal operation. This approach follows an unsupervised generative modeling concept where the model learns the individual characteristics of subcomponents (sensors/actuators) and the causal relationships among them under nominal condition, from a dataset. Then during regular operation, if a fault occurs in the system, it manifests itself as a low probability or anomalous event. Given an anomalous condition, further analysis can be performed to isolate which individual characteristics or causal relationship has changed to cause the anomaly. This provides a mechanism to perform root-cause analysis without using explicitly labeled training datasets for different faults. Thus, this approach potentially has good coverage, such that a single model can be leveraged for the detection and root cause isolation of multiple types of faults (even those that are previously unknown) in a CPS. Training such models is also easy as it avoids the extremely challenging task of acquiring sufficient labeled data for all types of faults in a CPS. Other benefits include the ability to handle heterogenous data, while accounting for the differences in the time scales for cyber and physical entities. While some studies in literature applied Bayesian networks (primarily in a supervised manner) for cyber security problems, our work applies Bayesian networks in an unsupervised manner for cyber-physical security problems.

BACKGROUND

This section provides a brief background and survey on Probabilistic Graphical Models (PGMs), with a focus on Bayesian networks and their applications to cyber security analysis.

Probabilistic Graphical Models (PGMs)

PGMs provide a succinct mechanism to model the joint distribution of statistically dependent random variables. The main benefit of PGMs is the ability to represent the joint distribution as a graph, which allows one to draw inferences about the underlying system without even knowing the parametric form of the model. Typically, a random variable is denoted as a node in the graph, while statistical dependencies are represented as edges (undirected or directed) between nodes. A Bayesian network is a type of PGM that allows one to capture causal information (cause and effect) using directed edges. Each node defines a conditional distribution of itself, given the parent nodes. The directionality of the edges are such that no directed cycles are induced in the overall graph. Hence, Bayesian networks are considered as directed acyclic graphs (DAGs). The overall joint distribution of the network is computed as a product of the conditional distribution defined by every node in the network.

Learning and *Inference* are the two main problems associated with Bayesian networks. The former involves learning the structure (the DAG) and the parameters of the conditional probability distribution. The goal is to identify the structure and the associated parameters that best explain the given data. Finding the optimal Bayesian network structure is a NP-hard problem, but efficient algorithms are available that often yield near optimal solutions (e.g. [5]). Bayesian networks support learning in supervised as well as in unsupervised settings, and thereby can be used with both labeled and unlabeled datasets. The second problem of inference pertains to finding probabilistic answers to user specified queries. For example, a user may seek the joint distribution of a subset of random variables given the observed values of another disjoint subset of random variables. Since, Bayesian networks only encode node-wise conditional probabilities, finding answers to such queries is not straightforward. However, efficient algorithms exist that allow one to find the exact answer to an arbitrary query using a secondary structure (such as junction tree) and a message-passing architecture [6]. Anomaly detection and root-cause isolation can both be interpreted as inference problems. In later sections, we show how the junction tree based inferencing can be leveraged for scalable root cause isolation.

Cyber Security Analysis using Bayesian Networks

Graphical security models are useful for visually representing and analyzing vulnerabilities in a system. Threat trees and Bayesian networks are two of the well-known graphical formalisms for security modeling [7]. Bayesian networks are versatile in that they can be constructed from attack models and domain knowledge, or learned from data. Attack graphs model how multiple vulnerabilities can be combined to result in an attack. Bayesian attack graphs combine attack graphs with computational procedures of Bayesian networks [8]. Wang et al. propose a probabilistic security metric for nodes in an attack graph and provide an algorithm for computing this metric in an attack graph [9]. Frigault et al. [10] provide a method to assign conditional probability to nodes in a Bayesian attack graph based on Common Vulnerability Scoring System scores (CVSS) and use that to calculate security metrics. They later extend their work to dynamic Bayesian networks to account for the evolving nature of vulnerabilities and availability of software patches [11]. Likewise, Houmb et al. quantify security risk level from CVSS

estimates of frequency and impact using Bayesian networks [12]. A Bayesian network modeling approach for separating different sources of uncertainty, such as uncertainty in attacker actions and attack success, for real-time security analysis is described in [13]. Feng and Xie provide an algorithm for merging expert knowledge and information stored in databases into a single Bayesian network [14]. PGMs have also been successfully used for root-cause analysis in different domains. For instance, Bayesian networks have been used for fault isolation in electrical power system [15], automotive systems [16], telecommunication networks [17] and manufacturing processes [18].

While the references cited above illustrate the use of graphical models for security analysis in different domains, we are not aware of any previous work that has developed Bayesian network models for anomaly detection and root-cause analysis in a cyber-physical system based on unlabeled data. We first formulate the challenge problem in the following section and then describe our technical approach to solve the problem.

PROBLEM FORMULATION

A cyber-physical system is a distributed system in which the sensors, actuators and controllers that are part of the physical world, coordinate their operation over a communication network. An insecure communication network makes the physical system vulnerable to different cyber attacks, which may adversely affect the operation of the system. Hence, in a cyber-physical system, anomalies in the physical state may arise either due to faults in the physical devices or due to malicious cyber attacks. Our goal is to first detect anomalies in the physical system and then determine the cause of the anomalies - whether a cyber attack or a physical fault was the most likely cause of the anomaly. This root-cause analysis is important for determining how the anomaly should be handled. If the anomaly is a result of a cyber attack, the IT staff or cyber professionals should be notified, so that appropriate security measures can be incorporated to prevent future attacks. On the other hand, if the root-cause analysis attributes the cause of the anomaly to a physical fault, then the control operator has to be notified, so that the faulty device can be fixed or replaced. Thus, an accurate anomaly detection and root-cause analysis approach enables the cyber-physical anomalies to be handled in a responsive manner.

System Description

The cyber-physical system that we use for our study is a building zone (which may consist of one or more adjacent rooms) that is instrumented with networked sensors and actuators to control the heating, ventilation, and air conditioning (HVAC) of the zone. These sensors and actuators communicate with a building automation system, using BacNet, a data communication protocol [19]. All physical devices on BacNet are assigned unique identifiers. The BacNet protocol can be used to remotely query or read the state of BacNet devices. BacNet can also be used to remotely write or modify the values of the actuators on the BacNet network at a certain priority level. In our building system, the sensors and actuators are configured to periodically report their states to the building automation system, which maintains a timestamped log of the values. The logs from the building automation system are supplied as input to the data-driven approach for learning the Bayesian network structure of the physical system.

Cyber Attack Mechanisms and Physical Faults

The networked building system described above is vulnerable to different types of attacks, since the BacNet protocol currently does not provide strong authentication mechanisms. An adversary may launch data integrity attacks remotely by sending erroneous sensor measurements and estimates or by setting incorrect actuator values. Such data integrity attacks affect the operational goals of the building system and render the information untrustworthy. A confidentiality attack results in unauthorized users gaining access to information about the physical parameters. A denial-of-service (DoS) attack can be launched by flooding the communication channels of the building system. In this work, we primarily focus on data integrity attacks that are launched from BacNet. We capture the BacNet traffic to the building sensors and actuators using Wireshark, which is a network sniffer. The network logs indicate which values were queried or written.

The physical faults in a building HVAC system can occur in the form of malfunctioning actuators, e.g., leaky water valves and stuck air dampers. Such physical anomalies are induced through electronic actuator override mechanisms for this current study.

Data Description

We selected the relevant fields from the Wireshark network logs as well as the logs of the physical system to generate the cyber-physical dataset. This time-series dataset of about 2500 records contains data corresponding to two cyber variables (which we generically refer to as *Cyb1* and *Cyb2*), two actuator variables (*Act1* and *Act2*) and four sensor variables (*Sense1*, *Sense2*, *Sense3*, and *Sense4*). Cyb1 is a BacNet issued identifier that uniquely identifies a sensor or actuator in the building system within the BacNet network and Cyb2 identifies the BacNet operation performed on the actuator or sensor. The actuator variables, Act1 and Act2 control heating and air flow to the zone respectively. Finally, the four sensors Sense1, Sense2, Sense3, and Sense4 monitor zone parameters, such as temperature and air flow at different building locations.

Cyb1 and Cyb2 are discrete variables that represent the cyber part of the building CPS. The cyber data collection is event-

Cyber	Physical
1	60
2	60
	62
	65
1	65

(a) Unaligned data.

Cyber	Physical
1	Δ = 0
2	Δ = 5
1	Δ = 0

(b) Event-based alignment.

Cyber	Physical
1	60
2	60
2	62
2	65
1	65

(c) Time-based alignment.

Figure 1. ALIGNMENT OF CYBER AND PHYSICAL DATA

driven, which means that there is a timestamped record in the Wireshark logs only when there is a read or write event on the BacNet. Wireshark provides timestamps at a nanosecond granularity. On the other hand, the remaining six variables listed above are continuous variables that represent the state of sensors and actuators in the physical system. This data is collected periodically every second and recorded in the building automation system logs. The different modalities of the cyber and physical data necessitate some data transformations, so that we can get an integrated cyber-physical dataset that is amenable to the data-driven approach.

One of the data transformations is the alignment of the cyber data with the physical data. When an actuator value is modified over BacNet, it may take several seconds for the modification to have an impact on the physical system, as shown in Figure 1(a). In this figure, a value of 1 in the cyber column shows a read operation, whereas a value of 2 indicates a write operation. The first record shows that at the time of the read event, the value of one of the physical variables is 60. Subsequently, when a write event occurs, it takes 3 seconds for the value of the physical variable to stabilize to a value of 65. While the cyber data is event-driven, the physical data is collected every second. Hence, although there is only one explicit cyber write event in the logs, the impact of that event on the physical state is recorded over multiple units of time. Therefore, we need a way to relate this temporal evolution to the corresponding cyber event. One way to achieve this is by aligning the cyber event with the changes it produces on the physical variables, instead of their raw values, as shown in Figure 1(b). However, when there are multiple events taking place concurrently, it may not be feasible to isolate these changes. Hence, we use time-based alignment, in which the gaps in the cyber events are filled by making the implicit cyber events explicit, as shown in Figure 1(c).

METHODOLOGY

We now layout the process for anomaly detection and root-cause isolation, which begins with learning a Bayesian Network from a given training set that contains data from nominal operation of the CPS system. The resulting Bayesian network characterizes the normal operation and hence, is capable of detecting anomalies as low probability events. The same Bayesian network also enables isolation of the root-cause of the detected anomaly. In the following sections we describe the aforementioned steps in greater detail.

Learning Bayesian Networks

In addition to aligning the time of the cyber events to the physical state, as discussed in the previous section, we also discretized the continuous variables. This data transformation is needed, because our learning process is based on discrete Bayesian networks. To this end, we used the discretization policy proposed in [20], that automatically determines the optimal number of bins and their widths, given the multivariate distribution of the variables. After discretizing that data, we learned a network structure (Directed Acyclic Graph) that maximizes the likelihood of observing the training data. As mentioned earlier, finding such a DAG is an NP-hard problem, hence we used efficient heuristics to approximate the underlying structure. It is important to penalize dense structures as they typically lead to over-parameterization and hence, over-fitting (bias-variance tradeoff). To address this tradeoff, we track the Bayesian Information Criterion (BIC) to drive our search for the best DAG. Figure 2 shows the Bayesian Network structure that was learned with the help of the GeNIe tool [21], using the building dataset that was described in the previous section. The thickness of an edge between a pair of nodes reflects the degree of statistical dependency between those nodes. For example Act2 has a very strong impact on Sense2. Hence, in Figure 2, we see that the edge connecting the two nodes is very thick. It should be noted that learning the parameters (conditional probability tables) is done as part of the structure learning process and need not be carried out separately.

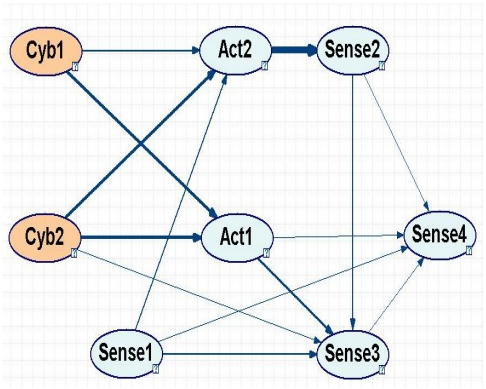

Figure 2. BAYESIAN NETWORK STRUCTURE BASED ON THE BUILDING CPS DATASET

Anomaly Detection

As noted earlier, given a Bayesian Network model of a CPS, a low probability event (with respect to the network) can be regarded as an anomaly. Let $N = \{N_1, N_2, \ldots, N_k\}, k \geq 1$ be the set of nodes in the Bayesian network. Let us define sets X and Y, such that $X, Y \subset N$ and $X \cap Y = \emptyset$. We refer to X and Y as target set and evidence set, respectively. We can estimate the conditional probability distribution of nodes in X, given the observed values of the nodes in Y i.e. $P_{X|Y}(X)$. Let $X(t)$ and $Y(t)$ denote the observed state of nodes in the respective sets at instant t. We define *anomaly score* (AS) with respect to $X(t)$ and $Y(t)$, as shown in (1), which quantifies the degree of deviation of an observed state from its most likely state. We use this metric for the purpose of anomaly detection and root cause isolation, by appropriately defining X and Y.

$$AS(X(t), Y(t)) = -log \frac{P_{X|Y(t)}(X(t))}{max(P_{X|Y(t)})} \quad (1)$$

In (1), $P_{X|Y(t)}(X(t))$ is the posterior likelihood of the observed target state at instant t, and $max(P_{X|Y(t)})$ is the likelihood of the most probable target state, given the evidence set $Y(t)$. The posterior distribution is obtained using the Junction Tree algorithm [6], which enables efficient computation of arbitrary joint posteriors in Bayesian Networks. The intuition behind (1) is that, as the observed state deviates away from the most probable state, the value of the anomaly score increases.

For anomaly detection we restrict the target set to consist of a single node, which can be either Act1 or Act2. Let T denote the target node. The evidence set includes all the nodes except the target node and is denoted as $N_{\backsim T}$. Then the anomaly score for a data sample at instant t is given by $AS(T(t), N_{\backsim T}(t))$. When this score exceeds a specified threshold, we classify that observation as an anomaly. Table 1 summarizes the results of anomaly detection for the building CPS dataset, using our Bayes network model and anomaly score metric. Anomalous readings due to cyber attacks on Act1 and Act2 were detected with an average accuracy of 81%. Anomalous readings in Act1 and Act2 due to physical faults were detected with an average accuracy of 78%.

Table 1. ANOMALY DETECTION ACCURACY

Type of anomalies	Accuracy
Cyber attacks (Act1, Act2)	81%
Physical fault (Act1, Act2)	78%

Root-Cause Isolation

The next step after detecting an anomalous event in a CPS is to identify its root cause, i.e. whether the anomaly was caused by a cyber attack or a physical fault. In general, a higher anomaly score need not indicate that the target node is the root cause. During this inferencing step, we leverage the posterior distributions computed by the Junction Tree algorithm during the detection phase. Note that the posterior distribution in (1) is a conditional distribution. If an error exists in any of the conditioning nodes, it will lead to a higher anomaly score. Therefore, disambiguating between different nodes to isolate the root cause using an exhaustive search is, in general, a combinatorial problem. Given a target node, our scalable solution limits the search for the root-cause to the nodes present in the Markov blanket of the target node. The Markov blanket of a node is defined as a set containing its parents and children and all the other parents of its children. This reduction in search space is possible due to the fact that the posterior distribution of a node is independent of the nodes outside its Markov blanket [6], provided the Markov blanket is fully observed. In Figure 3, the red curve shows the nodes in Markov blanket of Act2 and the blue curve shows the Markov blanket of Act1, for the Bayesian network given in Figure 2. The cyber variables appear in the Markov Blanket of both the nodes.

The primary goal of root-cause isolation is to rank the candidate nodes or "suspects" by how likely they are the cause of the detected anomaly and then designate the node that is the most likely, as the root-cause of the anomaly. To generate this ranked list, we introduce another metric called *root-cause potential (RCP)*. Let $T(t)$ denote the state of the target T at an instant t, which has been flagged as an anomaly by the anomaly detection phase. $RCP_{T(t)}(N_i)$ quantifies the likelihood that N_i is the root-cause of this anomaly. Then the node with the highest value of RCP among the candidate nodes is considered as the most likely root-cause of the detected anomaly. We present three algorithms for root-cause isolation. These algorithms define the RCP metric in different ways to generate the aforementioned ranked list.

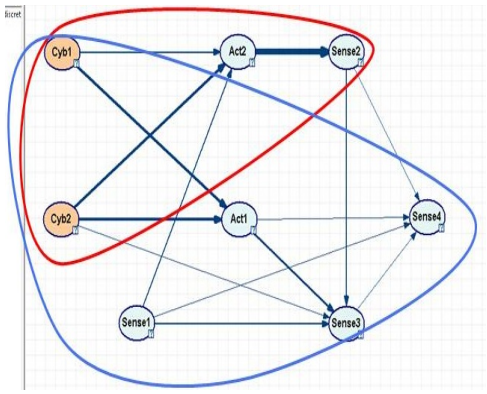

Figure 3. MARKOV BLANKET FOR THE CPS DATASET

While these algorithms use the anomaly score defined in (1) as a basis to compute the RCP of the nodes, they differ in their computational complexity.

Algorithm 1: Target Evidence

Input: B: Bayes net; T: Target node; Test data
Output: RC: Root Cause Node

1 $Y(t) = T(t)$
2 **foreach** $N_i \in MarkovBlanket(T)$ **do**
3 $\quad X(t) = N_i(t)$
4 $\quad RCP_{T(t)}(N_i) \leftarrow AS(X(t), Y(t))$
5 **end**
6 $RC(T(t)) = \text{argmax}_{N_i} RCP_{T(t)}(N_i)$

Algorithm 1: Target Evidence. A simple way to perform root-cause analysis is by evaluating every node in the candidate set against the target node. In other words, we ask the question: Given the observed state of the target T as evidence, what is the likelihood that a node $N_i \in MarkovBlanket(T)$ is the cause of the anomaly? Algorithm 1 is a method to address this question. In Algorithm 1, the RCP of a node is given by its anomaly score with respect to the target node (Line 4 in Algorithm 1).

It should be noted that the evidence set doesn't change during the RCP computation of the candidate nodes. As a result, the Algorithm 1 has a low computational overhead because it requires only a single message passing step. However, on the down side, when computing the RCP of a node N_i, it considers only the pairwise interaction between N_i and the target node while ignoring information about the other nodes in the Bayesian network. In general, there may be dependencies between N_i and other nodes that need to be considered. Performing root-cause analysis by considering only partial interactions may result in lower accuracy.

Algorithm 2: All Evidence.

Input: B: Bayes net; T: Target; Test data
Output: RC: Root Cause Node

1 **foreach** $N_i \in MarkovBlanket(T)$ **do**
2 $\quad X(t) = N_i(t)$
3 $\quad Y(t) = \backsim N_i(t)$
4 $\quad RCP_{T(t)}(N_i) \leftarrow AS(X(t), Y(t))$
5 **end**
6 $RC(T(t)) = \text{argmax}_{N_i} RCP_{T(t)}(N_i)$

Algorithm2: All Evidence. Algorithm 2 addresses the key shortcoming of Algorithm 1, by taking into account the interaction among all the nodes, instead of the just the target node T, when computing the RCP. This algorithm asks the following query: What is the likelihood that node N_i is the root-cause of the anomalous state $T(t)$, given the evidence about the other nodes in the candidate set excluding N_i?

The advantage of Algorithm 2 is that it uses more information than Algorithm 1 to perform root-cause analysis. However, the evidence set Y changes while computing the RCP of every candidate node (Line 3 in Algorithm 2). As a result, the message-passing step is repeated for every RCP computation, which leads to a higher computational overhead during inferencing.

Algorithm 3: Clique-Based Evidence. Algorithm 3 addresses some of the drawbacks of Algorithm 1 and Algorithm 2 to provide a more scalable way to perform root-cause analysis. Algorithm 3 uses more information than Algorithm 1 to infer the root-cause, but unlike Algorithm 2, it requires only a single message passing step. Instead of computing the RCPs of individual nodes, Algorithm 3 is a novel approach that computes the RCP of a collection of nodes, called cliques. These cliques are inherently generated during the junction tree based inference process. Of all the cliques present in the junction tree, the algorithm considers only those that contain at least one node in Markov Blanket of T. Unlike the previous two algorithms where the RCPs were computed using non-empty evidence sets, this algorithm uses an empty evidence set (Line 4 in Algorithm 3), and therefore requires only a single message passing step to compute the RCP of all the candidate nodes. The RCP of a node N_i is computed as the average of the anomaly scores of the cliques, Q_j, that N_i is a member of (Line 13 in Algorithm 3).

The low computational overhead of Algorithm 3 makes it amenable for online analysis. The cliques can be pre-computed

Algorithm 3: Clique-Based Evidence

Input: B: Bayes net; T: Target node; Test data
Output: RC: Root Cause Node

1 Determine the cliques $Q = \{Q_1, Q_2, \ldots, Q_k\}$ of B by running the junction tree inferencing algorithm.
2 $M(j,i) \leftarrow 1 \forall$ nodes $N_i \in$ clique Q_j
3 $M(j,i) \leftarrow 0 \forall$ nodes $N_i \notin$ clique Q_j
4 $Y(t) = \emptyset$
5 **foreach** $N_i \in MarkovBlanket(T)$ **do**
6 $\quad wtSum \leftarrow 0$
7 $\quad numCliq \leftarrow 0$
8 $\quad$ **foreach** $Q_j \in Q$ **do**
9 $\quad\quad X(t) = Q_j(t)$
10 $\quad\quad wtSum \leftarrow wtSum + M(j,i) * AS(X(t), Y(t))$
11 $\quad\quad numCliq \leftarrow numCliq + M(j,i)$
12 $\quad$ **end**
13 $\quad RCP_{T(t)}(N_i) \leftarrow \frac{wtSum}{numCliq}$
14 **end**
15 $RC(T(t)) = \text{argmax}_{N_i} RCP_{T(t)}(N_i)$

and as new data streams in, only the anomaly score and RCP needs to be computed. We have empirically observed that the smaller cliques lead to better isolation of the root-cause. A dense Bayesian Network, results in large cliques, therefore, in such cases, it may be necessary to consider the subsets of the clique, in order to achieve better root-cause isolation.

RESULTS AND DISCUSSION

We implemented the three root-cause analysis algorithms described in the previous section in Matlab and compared their performance using a test dataset from the building system. In this section, we present the results of the evaluation.

Root-Cause Evaluation

After detecting the anomalies in the anomaly detection phase, we determined the root-cause of different anomalies, using the 3 algorithms. Table 2 compares the accuracy of these algorithms. As mentioned earlier, the anomalies in our test data are due to either a cyber attack on the actuator variables controlling the heating (Act1) and air flow (Act2) or due to a fault in a physical variable that impacts the state of the Act1 and Act2 variables. We know the ground truth for the test dataset. Our goal is to determine the accuracy of the 3 algorithms by validating their inference with the ground truth.

The results show that when there is a cyber attack on Act1 (Column 2 in Table 2), Algorithm 1 is able to correctly infer that the root-cause of the anomaly is indeed a cyber variable, for only 2% of the test records. On the other hand, Algorithm 2 and Algorithm 3 have a much higher accuracy of 56% and 67%, respectively. When there is a cyber attack on Act2 (Column 3), Algorithm 1 incorrectly infers the root-cause of the anomaly to be a physical fault for all of the test records. Again, Algorithm 2 and Algorithm 3 have a much higher accuracy of 80% and 94%, respectively.

When the anomaly in Act1 is due to a fault in a physical variable (Column 4), Algorithm 1 correctly identifies the root-cause as a physical fault for 49% of the test records, while the accuracy of Algorithm 2 and Algorithm 3 is 77% for both cases. Similarly, when the anomaly in Act2 is due to a fault in a physical variable (Column 5), Algorithm 1 correctly identifies the root-cause as a physical fault for 84% of the test records, while the inference accuracy of Algorithm 2 and Algorithm 3 is 99% for both cases.

Table 2. ROOT-CAUSE ACCURACY

Algorithm	**Cyber Act1**	**Cyber Act2**	**Physical Act1**	**Physical Act2**
Algorithm 1	2%	0%	49%	84%
Algorithm 2	56%	80%	77%	99%
Algorithm 3	67%	94%	77%	99%

Table 3. COMPUTATIONAL TIME PER TEST RECORD

Algorithm	**Time (millisec)**
Algorithm 1	90
Algorithm 2	379
Algorithm 3	84

Computational Time

Table 3 shows the total computational time for anomaly detection and root-cause analysis per test record. Algorithm 2 has a high computational time, because the evidence changes and hence, the message passing step during inferencing has to be repeated when computing the RCP for each candidate node. On the other hand, Algorithm 1 uses only one additional message passing step at the beginning (Line 2 in Algorithm 1). Algorithm 3 does not require any additional message passing, as it reuses the message passing that was done as part of the anomaly detection phase. Hence, both these algorithms have relatively low computational overhead.

Algorithm Comparison

In summary, Algorithm 1 has low computational complexity. However, its inferencing is mainly conditioned on the observed state of the target node and it does not consider the dependencies between the other nodes. Hence, its inferencing accuracy is relatively poor. Algorithm 2 has a better accuracy than Algorithm 1, because its inference is based on more evidence. However, there may be some overfitting, because its inference is conditioned on the observed states of all the nodes in the complement set. Among the 3 algorithms, Algorithm 3 has the lowest computational overhead and the best root-cause accuracy for all the anomalies. Not only was it able to correctly infer the root-cause as a cyber attack or a physical fault, but in the case of the physical fault, in many cases it was also able to correctly infer the exact physical variable that caused the anomaly. This result can be attributed to the fact that Algorithm 3 exploits a natural partitioning of the graphical model via cliques and it examines individual cliques in an isolated manner to discover local anomalies that may result in faults at the system level.

CONCLUSIONS AND FUTURE WORK

Anomaly detection and root-cause analysis are essential for building resilient cyber-physical systems. Anomalies in cyber-physical systems, such as a smart grid or a smart building instrumented with networked sensors and actuators, may arise due to malicious cyber attacks or operational faults in the physical devices. Accurately detecting the anomalies and isolating their root-cause is important for identifying appropriate reactive and mitigating measures. However, this task is extremely challenging, as certain cyber attacks and physical faults may have very similar signatures on the system. Furthermore, the impact of a cyber attack on the operation of a physical system may manifest itself only after some time. Hence, we need suitable methods to not only model the causal relations between the cyber and physical variables, but also to take into account this temporal behavior.

In this paper, we presented a Bayesian network approach for learning these causal relations from unlabeled data. One of the challenges of the problem is that cyber data is typically event-driven and discrete, whereas physical data from sensors and actuators is periodic and continuous. We briefly discussed the data transformations that we performed to deal with the heterogeneity, so that the integrated dataset can be used to learn the Bayesian network structure and parameters. We also presented scalable algorithms to detect anomalies and isolate their root-cause using the Bayesian network approach. The novel clique-based algorithm for root-cause analysis was particularly effective in isolating the root-cause of cyber and physical attacks. It uses the cliques generated from the junction tree algorithm for Bayesian networks to compute the correlations and isolate the root-cause.

As part of the future work, we plan to validate these Bayesian network algorithms for anomaly detection and isolation using larger datasets from different domains. In addition, a couple of key technical research directions currently being pursued are: (i) explicit modeling of temporal correlations among cyber and physical entities for increased robustness of data-driven diagnosis algorithms, and (ii) discovery of fault propagation patterns and attack graphs for a cyber-physical system from a generative Bayesian network model of the system.

REFERENCES

[1] Blanke, M., Kinnaert, M., Lunze, J., and Staroswiecki, M., 2003. *Diagnosis and Fault-Tolerant Control*. Springer-Verlag.

[2] Cardenas, A., Amin, S., Sinopoli, B., Giani, A., Perrig, A., and Sastry, S., 2009. "Challenges for Securing Cyber Physical Systems". In Workshop on Future Directions in Cyber-physical Systems Security.

[3] Mo, Y., Kim, T., Brancik, K., et al., 2011. "Cyber-Physical Security of a Smart Grid Infrastructure". *Proceedings of the IEEE*.

[4] Kosut, O., Jia, L., Thomas, R., and Tong, L., 2010. "Malicious Data Attacks on Smart Grid State Estimation: Attack Strategies and Countermeasures". In First International Conference on Smart Grid Communications.

[5] Chickering, D., 2002. "Optimal Structure Identification With Greedy Search". *Journal of Machine Learning Research,* ***3***, pp. 507–554.

[6] Koller, D., and Friedman, N., 2009. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

[7] Kordy, B., Cambacedes, L., and Schweitzer, P. DAG-based Attack and Defense Modeling: Don't Miss the Forest for the Attack Trees.

[8] Liu, Y., and Man, H., 2005. "Network Vulnerability Assessment using Bayesian Networks". In Proc. of SPIE Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security, Vol. 5812, pp. 61–71.

[9] Wang, L., Islam, T., Long, T., Singhal, A., and Jajodia, S., 2008. "An Attack Graph-Based Probabilistic Security Metric". In IFIP WG 11.3 Working Conference on Data and Applications Security, Springer-Verlag, pp. 283–296.

[10] Frigault, M., and Wang, L., 2008. "Measuring Network Security Using Bayesian Network-Based Attack Graphs". In Proceedings of IEEE Computer Software and Applications Conference.

[11] Frigault, M., Wang, L., Singhal, A., and Jajodia, S., 2008. "Measuring Network Security Using Dynamic Bayesian Network". In Proceedings of QoP.

[12] Houmb, S., Franqueira, V., and Engum, E., 2009. "Quantifying Security Risk Level from CVSS Estimates of Fre-

quency and Impact". *Journal of Systems and Software,* ***83***(9).

[13] Xie, P., Li, J., Ou, X., Liu, P., and Levy, R., 2010. "Using Bayesian Networks for Cyber Security Analysis". In Intl. Conference on Dependable Systems and Networks (DSN).

[14] Feng, N., and Xie, J., 2012. "A Bayesian Networks Based Security Risk Analysis Model for Information Systems Integrating the Observed Cases with Expert Experience". *Scientific Research and Essays,* ***7***(10), pp. 1103–1112.

[15] Choi, A., Zheng, L., Darwiche, A., and Mengshoel, O. *Data Mining in System Health Management*. ch. A Tutorial on Bayesian Networks for System Health Management.

[16] Jansson, M., 2004. "Fault Isolation Using Bayesian Networks". Master's thesis, KTH.

[17] Velasco, J. M., 2012. "A Bayesian Network Approach to Diagnosing Root Cause of Failure from Trouble Tickets". *Artificial Intelligence Research,* ***1***(2), December.

[18] Pradhan, S., Singh, R., Kachru, K., and Narasimhamurthy, S., 2007. "A Bayesian Network Based Approach for Root-Cause Analysis in Manufacturing Process". In Proceedings of Intl. Conference on Computational Intelligence and Security.

[19] Bacnet. www.bacnet.org.

[20] Sarkar, S., Srivastav, A., and Shashanka, M., 2013. "Maximally Bijective Discretization for Data-Driven Modeling of Complex Systems". In Proceedings of American Control Conference.

[21] Decision Systems Laboratory. GeNIe and SMILE. genie.sis.pitt.edu.